computeSquares

Method Breakdown
(for loops)

For Loop

```
    1        2      4
for (initialization; test; update) {
    statements;
}         3
```
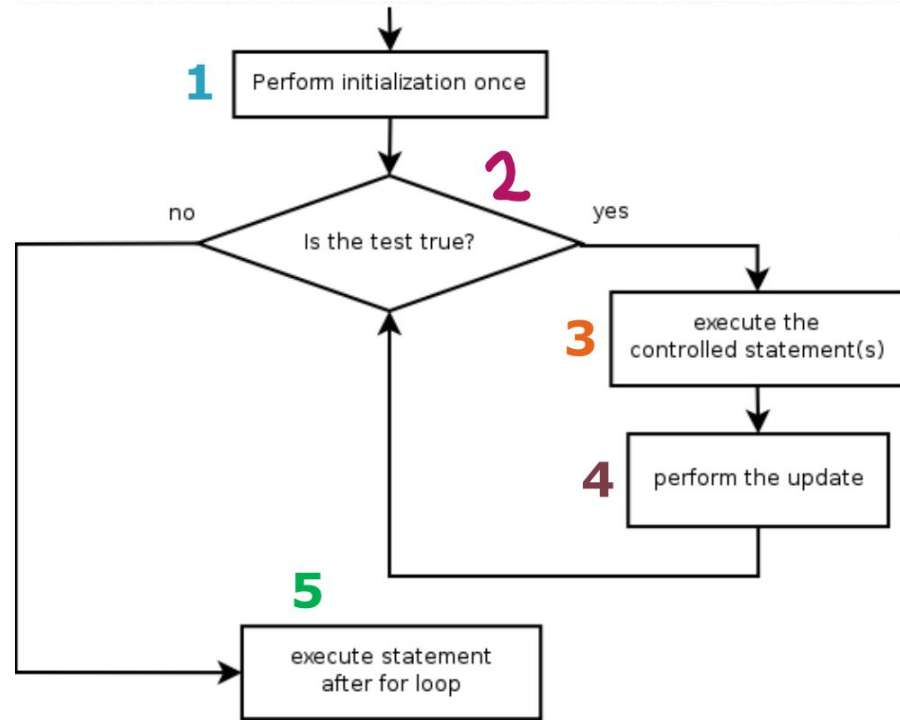**5**

Initialization: declare and initialize a loop variable

Test: an expression that will be evaluated to determine whether to continue the loop or end it

Update: update the loop variable to make progress toward the test failing

These 3 components determine exactly how many times the for loop will run

Statements in the body of the loop will be executed each time the test evaluates to true

Method Description

Write a method called computeSquares to produce the following text:

```
1 squared = 1

2 squared = 4

3 squared = 9

4 squared = 16

5 squared = 25

6 squared = 36
```

Algorithm: (step by step procedure)

for the numbers 1, 2, 3, 4, 5, 6:
    Print the number and it's squared value on a single line of console output in the specified format
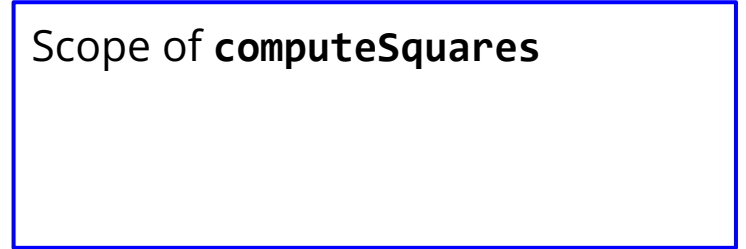
What we need:

❏   for loop with a variable going from 1 to 6 by +1
    ❏   println, String literal, loop variable, expressions

```
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
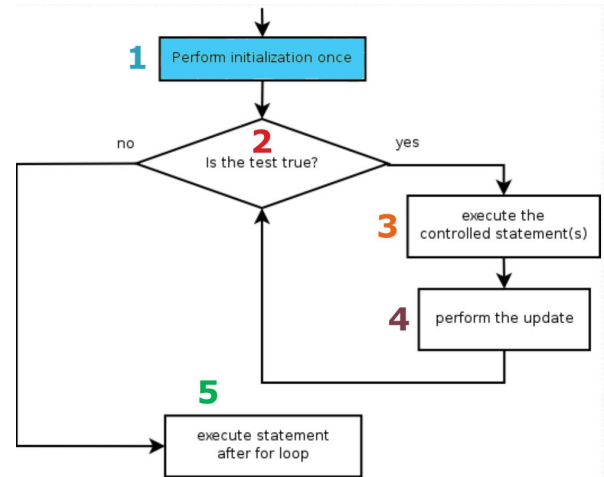
Output:

Scope of **computeSquares**

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
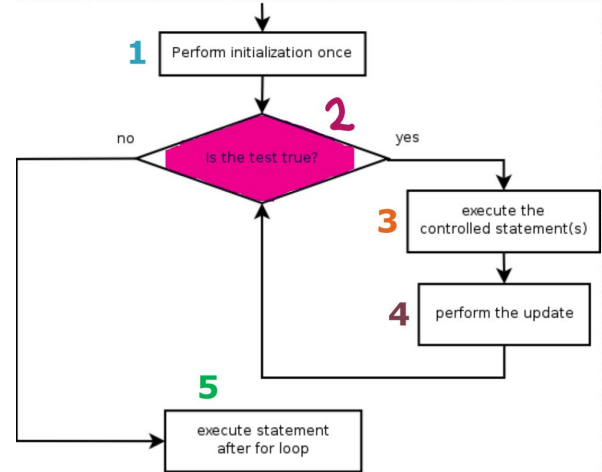
Scope of **computeSquares**

Output:

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Output:

Scope of **computeSquares**
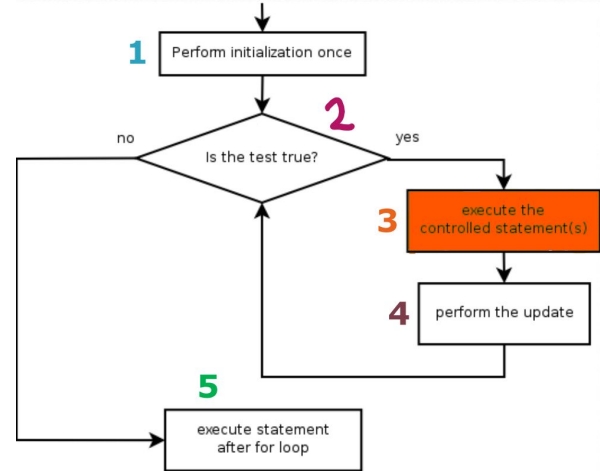
number	1

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	1

Output:

1 Perform initialization once

2 Is the test true?

no   yes

3 execute the controlled statement(s)

4 perform the update

5 execute statement after for loop

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
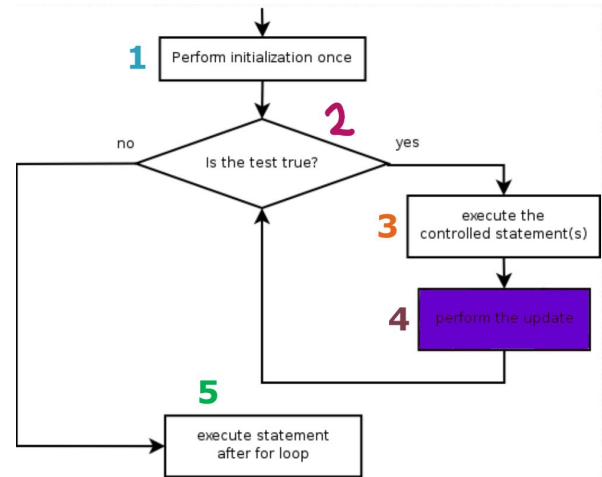
Output:

```
1 squared = 1
```

Scope of **computeSquares**

number	1

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
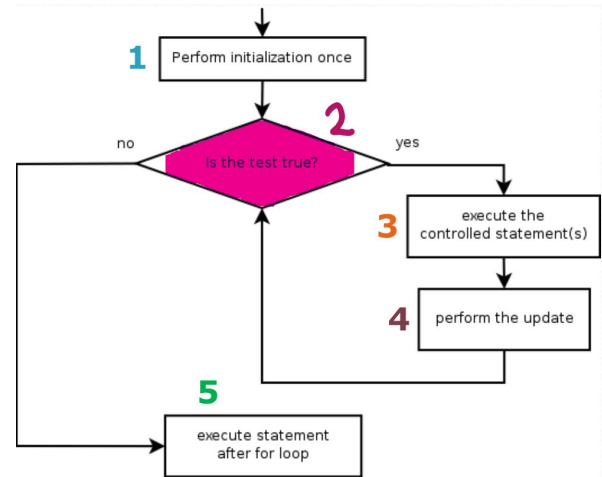
Scope of **computeSquares**

number	2

Output:

```
1 squared = 1
```

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	2

Output:

```
1 squared = 1
```

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
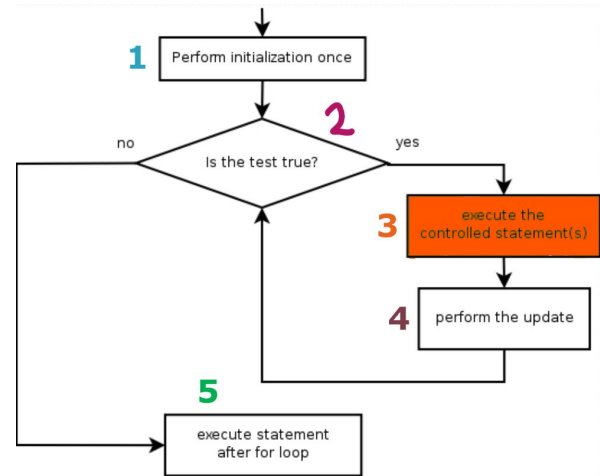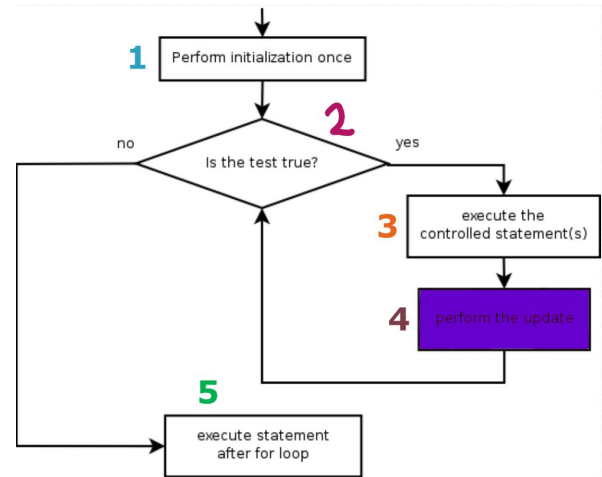
Output:

```
1 squared = 1
2 squared = 4
```

Scope of **computeSquares**

number	2

1 Perform initialization once

2 Is the test true?   no   yes

3 execute the controlled statement(s)

4 perform the update
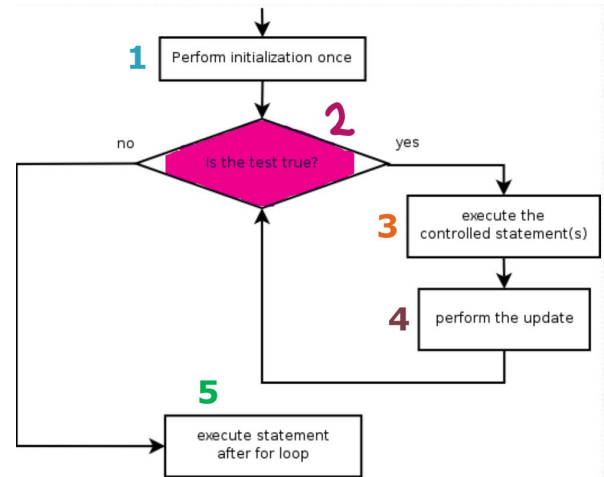
5 execute statement after for loop

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	3

Output:

```
1 squared = 1
2 squared = 4
```

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Output:

```
1 squared = 1
2 squared = 4
```

Scope of **computeSquares**

number	3

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
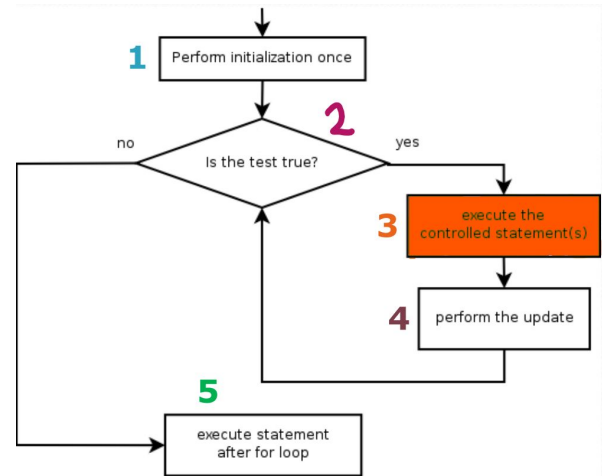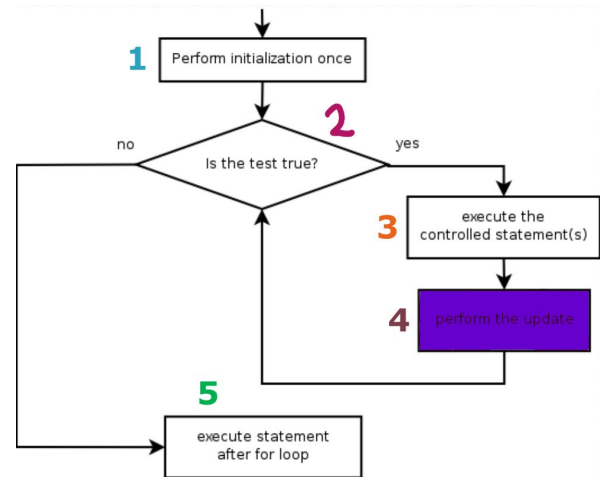
Scope of **computeSquares**

number	3

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
```
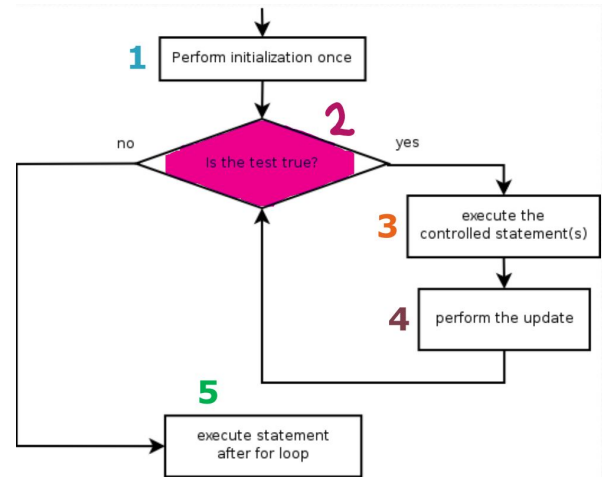
```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
```

Scope of **computeSquares**

number	4

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	4

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
```

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```
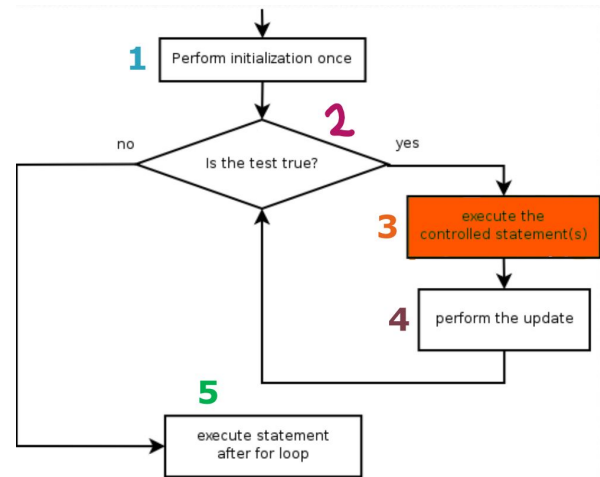
Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
```

Scope of **computeSquares**
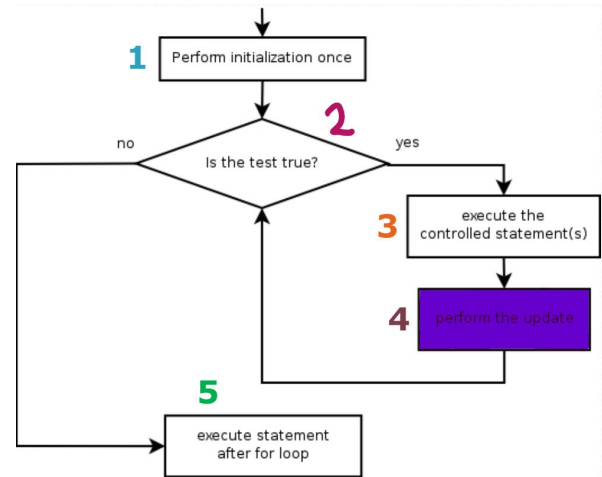
number	4

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	5

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
```
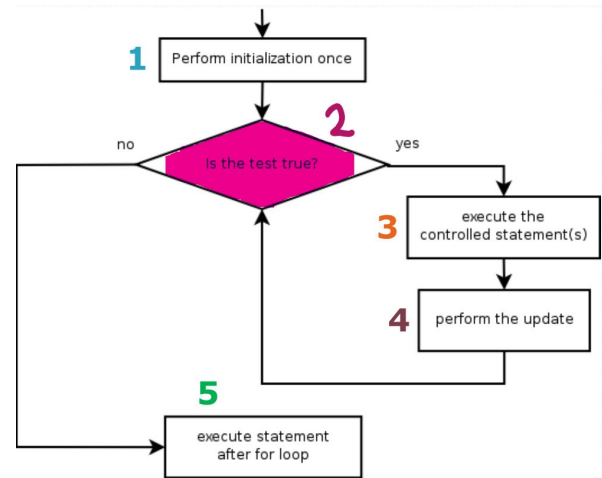
```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	5

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
```
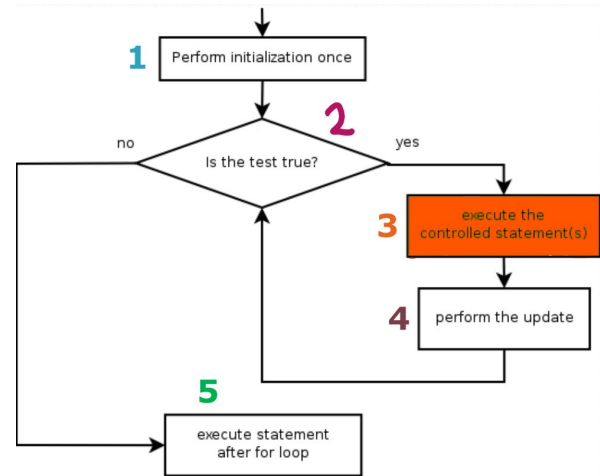
```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	5

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
```


1 Perform initialization once

2 Is the test true?

no   yes

3 execute the controlled statement(s)

4 perform the update
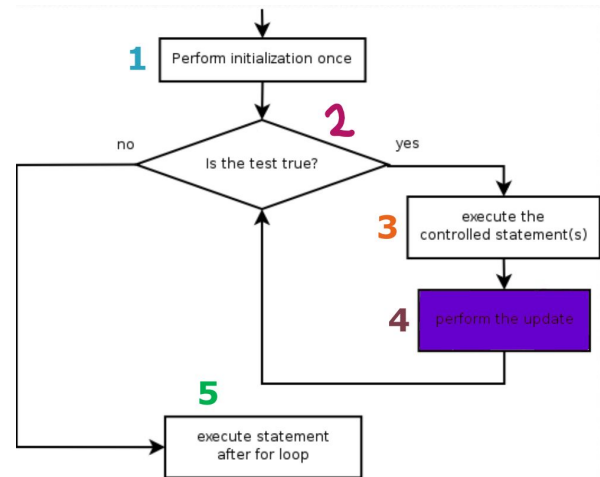
5 execute statement after for loop

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
```

Scope of **computeSquares**
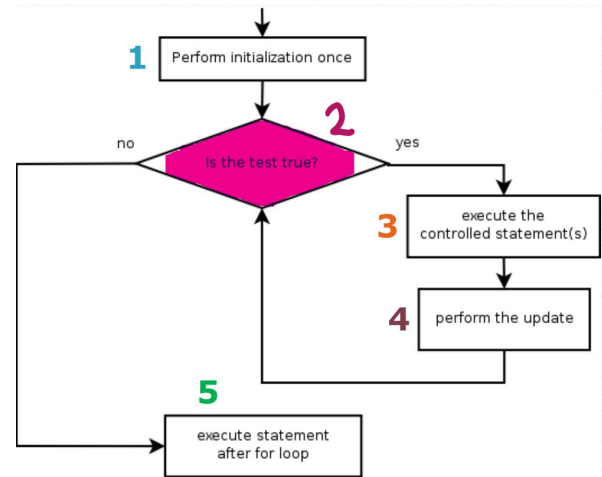
number	6

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	6

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
```
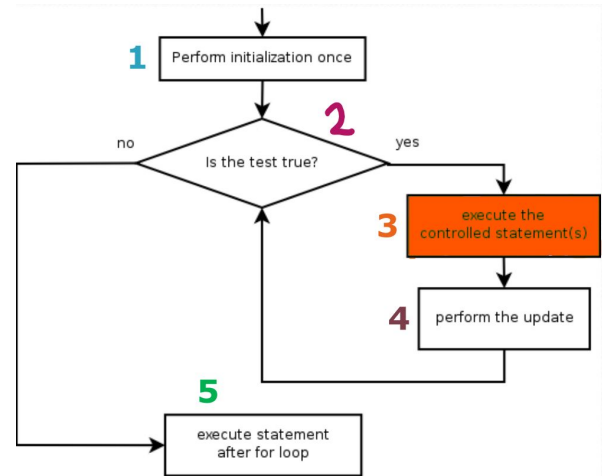
```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

number	6

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
6 squared = 36
```


1 Perform initialization once

2 Is the test true?

no   yes

3 execute the controlled statement(s)

4 perform the update
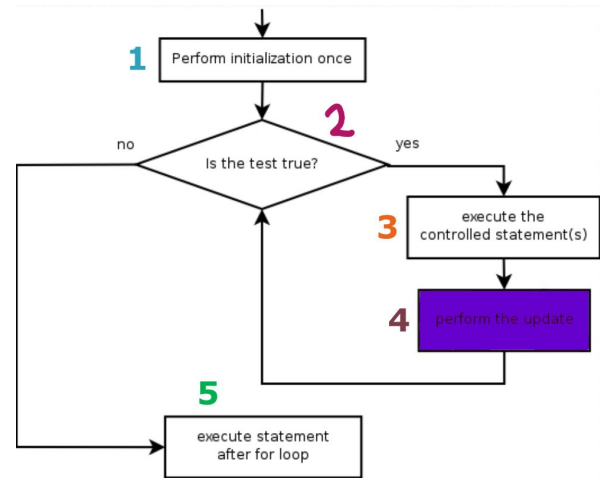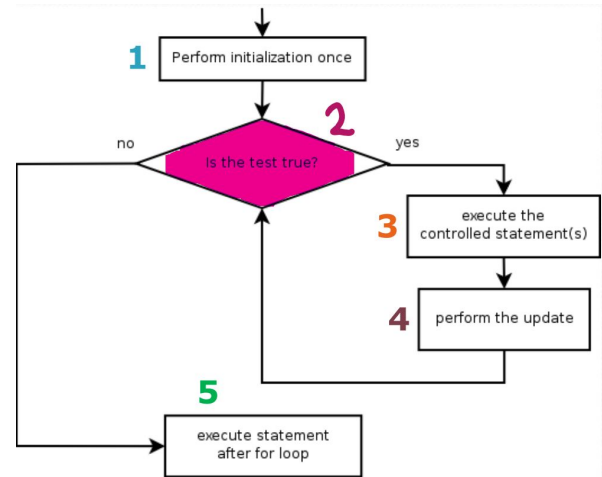
5 execute statement after for loop

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
6 squared = 36
```

Scope of **computeSquares**

number	7

1   Perform initialization once

2   Is the test true?   no / yes

3   execute the controlled statement(s)

4   perform the update

5   execute statement after for loop

```java
public static void computeSquares() {
    for (int number = 1; number <= 6; number++) {
        System.out.println(number + " squared = " + (number * number));
    }

}
```

Scope of **computeSquares**

Output:

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25
6 squared = 36
```